

Automated Healthcare Advisor: A PDF-Driven LLM-Powered Recommendation System

*Note: Sub-titles are not captured in Xplore and should not be used

1st Rajkumar Gajera
CSE Department
Nirma University
Ahmedabad, India
r.c.gajera rg@gmail.com

Abstract—This paper presents the Automated Healthcare Advisor, a hybrid Retrieval-Augmented Generation (RAG) system that combines PDF document processing with Large Language Models (LLMs) to deliver context-aware, real-time recommendations. The system utilizes FAISS-based semantic vector search for efficient information retrieval and employs LLaMA2 for coherent and relevant response generation. A key innovation of this work is the integration of a blog-based dynamic knowledge repository, which augments static PDF content with up-to-date web information, enhancing both the breadth and contextual accuracy of the system’s knowledge base. Experimental evaluations demonstrate significant improvements in multi-document retrieval performance, semantic chunk processing, and response coherence. The proposed architecture offers a scalable and adaptable framework for complex document interaction tasks, with applications across healthcare, legal, and educational domains.

Index Terms—Large Language Models (LLMs), Semantic Search, FAISS, LLaMA2, Natural Language Processing (NLP), Hybrid AI Systems, Context-Aware Recommendation, Information Retrieval.

I. INTRODUCTION

The exponential growth of digital documentation—particularly in Portable Document Format (PDF)—has fundamentally transformed information dissemination in sectors such as healthcare, law, education, and business. PDFs are widely adopted due to their portability, formatting fidelity, and multimedia embedding capabilities. However, their unstructured layouts, absence of semantic tagging, and non-linear reading order present significant obstacles to information extraction [1], [2]. These issues are further compounded when users require accurate, real-time, context-aware answers rather than keyword-matched document fragments.

Traditional document retrieval systems, built primarily on lexical keyword matching, fall short in addressing complex, ambiguous, or domain-specific queries. They typically operate on static data silos, offering limited capabilities for semantic reasoning, multi-document comparison, or live updates. This gap renders such systems inadequate in domains where knowledge evolves rapidly and semantic interpretation is critical [3].

Recent advancements in Natural Language Processing (NLP), particularly through Retrieval-Augmented Genera-

tion (RAG) and transformer-based Large Language Models (LLMs), provide powerful alternatives. Models such as LLaMA 2 [4], when coupled with dense vector search backends like Facebook AI Similarity Search (FAISS) [5], enable real-time, intent-aware querying across unstructured documents. These systems go beyond shallow retrieval, incorporating vector embeddings to capture contextual relationships between questions and document segments.

To further enhance relevance and precision, state-of-the-art systems incorporate structure-aware chunking and advanced query rewriting techniques. Chunking strategies now preserve document hierarchies, figures, and tables, improving both retrieval granularity and semantic fidelity [6]. Meanwhile, transformer-based query rewriters like FLAN-T5 [7] dynamically reframe ambiguous or underspecified queries into more precise sub-questions—significantly boosting retrieval performance across complex datasets.

This paper introduces a novel, hybrid intelligent document chatbot architecture designed to bridge the gap between static PDF processing and dynamic knowledge augmentation. The system integrates offline documents with a continuously updating online repository, enabling real-time, high-recall, semantically coherent responses.

A comparative overview of traditional systems and the proposed architecture is presented in Table I, highlighting differences in query handling, semantic understanding, chunking, knowledge integration, and scalability.

Key innovations include:

- **Semantic Chunking and Tokenization:** Structure-preserving chunking (e.g., heading-based and visual segmentation) ensures document integrity while enabling token-efficient LLM processing [6].
- **Hybrid Knowledge Base:** Static PDF content is augmented with real-time, trusted online content (e.g., blogs, wikis), significantly enhancing the temporal relevance and completeness of responses.
- **Query Rewriting and Reranking:** Multi-intent and vague user queries are transformed into more precise forms using pretrained models like FLAN-T5, followed by

reranking using cross-encoders for improved accuracy [7], [8].

- **Modular and Scalable Architecture:** Built on FAISS for vector search and LLaMA 2 for response generation, the architecture supports real-time semantic search and future modular extensions for domain-specific tuning.

Feature	Existing Systems	Proposed System
Query Scope	Single-document, static retrieval	Multi-document, cross-source, real-time querying
Semantic Understanding	Shallow keyword matching	Deep contextual embeddings with LLMs
Chunking Strategy	Fixed-size, naïve segmentation	Structure-aware, layout-preserving chunking [9]
Knowledge Base	Static PDFs only	Static PDFs + Dynamic online knowledge (e.g., blogs)
Query Interpretation	Direct execution of raw queries	FLAN-T5-based rewriting and semantic reranking [10], [11]
Answer Coherence	Fragmented, document-centric	Multi-perspective, contextually linked responses
Scalability & Performance	Limited to small datasets	Real-time, FAISS-accelerated dense retrieval
Extensibility	Rigid, closed pipeline	Modular and customizable for domain-specific LLM tuning

TABLE I
COMPARISON OF EXISTING AND PROPOSED SYSTEMS

The system is applicable across multiple domains:

- **Healthcare:** Assists clinicians by integrating medical reports with real-time alerts, journal updates, and drug policy changes from reputable sources [12].
- **Legal Research:** Enables semantic search over statutes and case law with context-aware synthesis from legal commentary blogs and databases [13].
- **Education:** Offers learners multi-document comparison, summarization, and adaptive Q&A for textbooks and research articles.
- **Business Intelligence:** Supports market analysts by correlating financial reports with real-time news, forecasts, and blog insights [14].

By addressing the limitations of traditional static search engines—such as shallow semantic understanding, limited context preservation, and inability to integrate dynamic external knowledge sources—the proposed architecture establishes a robust foundation for next-generation, real-time intelligent document systems. As highlighted in Table I, the system surpasses existing solutions across key dimensions including query interpretation, semantic chunking, hybrid knowledge integration, and scalability. By combining the reliability of offline PDF content with the agility of live web resources, the architecture facilitates nuanced, context-aware, and temporally relevant information access across multiple domains.

II. LITERATURE SURVEY

The implementation of chatbots that engage with PDF has been an active focal area because of enhanced NLP and

progress in ML. Other conventional document search systems that simply searched for the key words only proved difficult in handling unstructured data such as PDFs. In particular, NLP and transformer models make the application of such systems much more efficient in extracting and reading text from PDFs. These advancements have provided platform to develop intelligent document based chat bots to serve user queries.

Transformer models such as BERT and GPT have greatly improved the comprehension of context and semantic interpretation of references with relation to documents, in chatbots. In contrast to previous practices that are based on simple keyword matching, transformer models provide a better solution in the problem of distinctly meaning extraction from more complex data. Further, vector based search system like FAISS has also helped in the field of document search where rather than searching the exact match the search system can also search for related documents. Such systems are particularly helpful in sorting out large databases, which are prevalent in healthcare.

Adding OCR with deep learning models has also emerged as an important area of study for extract of text from the scanned PDFs. Despite these drawbacks in OCR technologies, main issues appear when the font type or document layout deviate from a standard form, hybrid systems which combine OCR and machine learning were reported to be 94% effective. Such systems are capable not only of text mining but also of categorizing the mined data – and this is a crucial advantage in terms of building a correct model of the document.

The implemented code incorporates various methodologies highlighted in the literature, combining them effectively for a robust PDF-based chatbot system. For example, [16] and [17] emphasize transformer-based models like BERT and GPT-3 for contextual understanding and response generation. The code uses Llama2 through CTransformers to achieve similar goals while optimizing computational efficiency, addressing the limitations of resource-intensive models mentioned in [17]. [19] integrates OCR with NLP for scanned documents, but this implementation focuses on pre-extracted text from PDFs, avoiding challenges such as OCR errors discussed in [19].

The use of FAISS in the implementation aligns with methodologies in [10], [21], and [22], which focus on efficient vector-based searches and LLM integration. This system builds on these concepts by incorporating HuggingFace embeddings for semantic text search, providing high-dimensional accuracy while addressing performance limitations in large datasets noted in [22]. Unlike [9], which tackles multi-modal data such as images and tables, the code centers on structured, text-rich PDFs, similar to the focus of [20].

Custom prompt templates and a retrieval QA chain ensure accurate contextual responses, paralleling the goals of [18] and [20], which aim to enhance user interaction and response quality. By leveraging LangChain for PDF management and query handling, the system further optimizes processes described in [20], creating a scalable solution for large document interactions. Deployment through Chainlit offers a lightweight approach compared to broader frameworks discussed in [18],

Ref No.	Methodology	Contribution	Focus Area	Performance Metrics	Limitations
[15]	Traditional information retrieval systems	Discussed the limitations of keyword-based search systems and laid the foundation for modern retrieval techniques.	Limitations of Keyword Matching	Retrieval accuracy, Relevance	Inability to handle semantic content and complex queries.
[16]	Transformer-based models (BERT)	Introduced BERT, improving document search and contextual understanding.	Contextual Understanding in NLP	Accuracy, F1-score	Requires large computational resources and training time.
[17]	Large-scale transformer models (GPT-3)	Demonstrated GPT-3's ability to generate contextually accurate responses to queries.	Generative Models for Question Answering	BLEU, ROUGE, Perplexity	High computational costs for training and fine-tuning.
[18]	NLP-based chatbot systems	Explored NLP techniques for developing chatbots that interact with documents, specifically PDFs.	NLP for Document-based Chatbots	Response Accuracy, User Satisfaction	Limited by the diversity of document types and structure.
[19]	OCR combined with NLP	Integrated OCR with chatbot systems to enable interaction with scanned PDF documents.	OCR and NLP Integration for PDF Interaction	OCR Accuracy, Query Answer Accuracy	OCR errors with low-quality scans and complex layouts.
[9]	Multi-modal machine learning models	Proposed ML models capable of handling multi-modal content, such as tables and figures within PDFs.	Multi-modal Document Interaction	Precision, Recall, F1-score	Difficulty in extracting data from non-text elements.
[10]	Vector-based search (FAISS)	Introduced FAISS for efficient vector-based search, enabling improved retrieval in PDF-based chatbot systems.	Efficient Semantic Search in PDFs	Recall, Mean Reciprocal Rank	Limited to text-based content retrieval, not images or tables.
[11]	Llama2 integration with PDFs	Combined Llama2 with document interaction to enhance chatbot responses and interaction with PDFs.	Enhancing Chatbots with Llama2 for PDF Interaction	BLEU, ROUGE, User Interaction Rate	Llama2's performance may degrade with noisy, inconsistent PDFs.
[20]	LangChain and NLP	Integrated LangChain in document-based chatbots to optimize the handling of large-scale PDFs.	LangChain Integration for PDF Handling	Response Time, Query Accuracy	Scalability issues for very large documents.
[21]	FAISS for high-dimensional search	Focused on FAISS for high-dimensional similarity search, making it easier to find related content in PDFs.	Efficient Similarity Search with FAISS	Precision, Search Speed	Limited by the dimensionality of the vector space.
[22]	FAISS and LLM integration	Investigated the integration of FAISS with large language models (LLMs) for real-time query answering from PDFs.	Combining FAISS and LLMs for PDF-based Query Answering	F1-score, Query Latency	Performance drops with larger datasets or complex queries.

TABLE II
STATE-OF-THE-ART APPROACHES TO AI-DRIVEN DOCUMENT QUERY SYSTEMS: METHODOLOGIES, METRICS, AND LIMITS

providing an efficient and focused interaction platform tailored to text-based PDF content.

To summarize, Table II presents a comparative analysis of key methodologies from the literature, focusing on their contributions, performance metrics, and limitations. This system draws from these approaches while addressing gaps in computational scalability, real-time performance, and domain-specific retrieval—thus delivering a specialized and efficient PDF-based document query platform.

III. PROPOSED METHODOLOGY

Our system leverages a modular, intelligent architecture tailored to provide accurate, real-time answers by integrating both offline documents and live online data sources. The complete workflow—from user interaction to final evaluated response—is visually illustrated in Fig. 1, offering a comprehensive overview of the components and data flows that underpin our system.

The process begins at the Chainlit interface, where users input natural language queries. These inputs are first routed through a Query Rewriting Module, which enhances the semantic precision of the query using a pretrained FLAN-T5 model. This rewritten query is then simultaneously dispatched to two content pipelines: the first handles PDF ingestion, extracting structured and unstructured data from documents, while the second employs web scraping to gather the most

recent and contextually relevant information from online sources. Both pipelines feed into the Embedding Creation

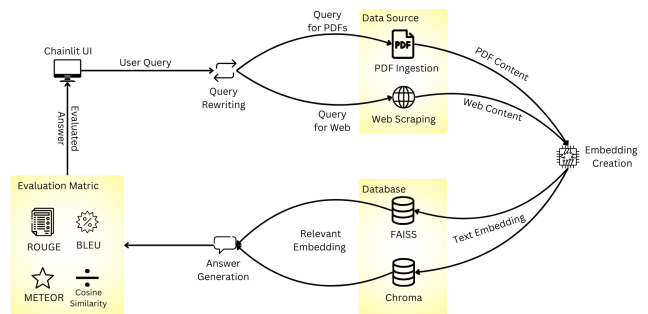


Fig. 1. Overall System Architecture

Module, where textual data is transformed into dense semantic vectors using the all-MiniLM-L6-v2 model. These embeddings are stored in a dual-database system designed for optimized retrieval: FAISS, for high-speed approximate nearest neighbor search, and Chroma, for hybrid retrieval combining semantic and lexical matching. This setup ensures robust access to contextually relevant information during downstream retrieval tasks.

The most relevant embeddings are passed to the Answer Generation Module, which synthesizes information using powerful open-source language models such as LLaMA 2 or Mistral (running locally via Ollama). This module generates well-formed, factually grounded responses by aggregating evidence from both web and PDF sources.

Before being returned to the user, the generated answer undergoes rigorous quality assessment in the Evaluation Module. This module employs a multi-metric strategy—including BLEU, ROUGE, METEOR, cosine similarity, and optional LLM-based scoring—to evaluate the linguistic fluency, factual accuracy, and semantic similarity of the answer. These metrics ensure comprehensive quality control and support iterative system improvement.

Figure 1 visually encapsulates this entire pipeline, showing how each module integrates seamlessly into the system. Furthermore, Table III summarizes the key tools, models, and hyperparameters used across each component, offering a concise reference for the system’s configuration and reproducibility.

Module	Model/Tool	Key Parameters
Embedding	all-MiniLM-L6-v2	dim=384, batch=64
Query Rewriting	flan-t5-base	max_length=64
Retrieval	FAISS + Chroma	top_k=2, hybrid=True
Response Generation	LLaMA2 / Mistral (Ollama)	temp=0.5, max_tokens=512
Evaluation	BLEU, ROUGE, METEOR	LLM-Judge: score 1-10

TABLE III
MODEL CONFIGURATIONS AND KEY PARAMETERS

Finally, the fully evaluated and verified response is delivered back to the user through the Chainlit interface, completing the query-to-answer loop. This tightly integrated system is designed for high responsiveness and accuracy, making it a scalable and reliable solution for delivering up-to-date information in demanding fields like medicine, technology, and academia.

A. Embedding Generation: Building the System’s Semantic Core

At the heart of our system lies the Embedding Generation module, which acts as its semantic backbone (see Fig. 2). This is where we translate raw text from PDFs and web pages into high-dimensional vector embeddings—a sophisticated, machine-readable format that captures the underlying meaning of the content. This allows our system to perform intelligent, context-aware searches that go far beyond simple keyword matching.

The process begins after the text has been extracted and cleaned. This refined content undergoes Document Chunking, where it is split into overlapping segments to preserve context across boundaries. These chunks are then fed into an embedding pipeline powered by the Hugging Face sentence-transformer model (all-MiniLM-L6-v2). The model converts

these chunks—such as paragraphs or grouped sentences—into compact, fixed-size vectors. The strength of these vectors lies in their ability to preserve semantic relationships; texts with similar meanings appear close together in the “embedding space,” even if their actual wording is different.

To enable fast and accurate retrieval, these embeddings are stored in two specialized vector databases:

- FAISS (Facebook AI Similarity Search): Used for dense embedding and indexing, it supports rapid semantic similarity searches, retrieving content that is meaningfully related to the user’s query.
- Chroma: Provides sparse indexing capabilities, enabling hybrid search that blends dense semantic search with traditional keyword-based retrieval, which is particularly beneficial in technical or medical documents.

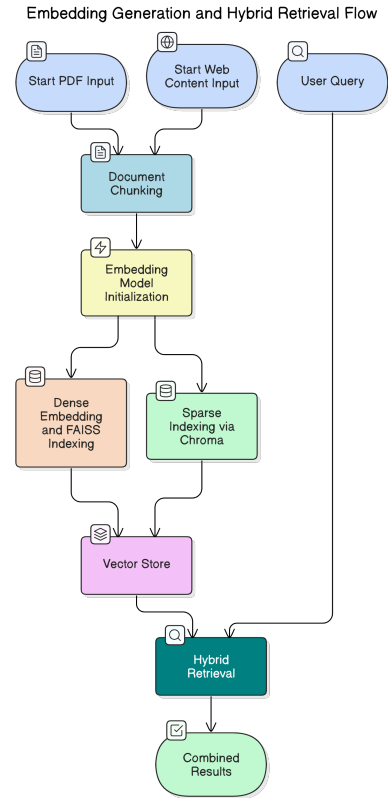


Fig. 2. Vector Database creation

This dual-indexing strategy, as illustrated in Fig. 2, ensures a robust and comprehensive hybrid retrieval pipeline. Additionally, every chunk is tagged with metadata such as the source file, page number, or URL, ensuring traceability of information. To further preserve context and reduce information loss due to LLM token limits, hierarchical chunking is applied—segmenting documents with slight overlaps.

B. User Query Processing and Retrieval

The User Query Processing and Retrieval module serves as the critical bridge between a user’s natural-language question

and our system’s structured knowledge base (see Fig. 3). Its core function is to transform potentially vague or underspecified user inputs into rich, semantically meaningful queries that can retrieve the most relevant information.

To accomplish this, our system applies a hybrid query rewriting strategy combining two complementary techniques:

- **N-gram Prediction:** A probabilistic language model—trained on the entire corpus—is used to analyze the structure of short user queries and expand them with likely succeeding terms. This yields more contextually rich, syntactically valid queries, increasing the chances of a relevant retrieval.
- **Transformer-Based Rewriting:** Simultaneously, we utilize the google/flan-t5-base model to rephrase the user input into a more precise and complete question. For example, an input like “heart issue treatment” is rewritten as “What are the available treatment options for common heart-related issues?”, which aligns more closely with the language and structure of the indexed data.

Both the N-gram-expanded and transformer-rewritten versions are stored as query variants and passed to the hybrid retrieval phase, which consists of:

- **Dense Retrieval using FAISS:** Retrieves semantically similar chunks based on embedding proximity.
- **Sparse Retrieval using Chroma:** Retrieves chunks that match specific keywords or domain-specific terms.

The results from both engines are combined and deduplicated, ensuring a high-quality, non-redundant pool of candidate documents. This fusion of dense and sparse methods ensures high recall (by capturing semantically relevant documents) and high precision (by including exact keyword matches). The unified result set is then refined into the Final Output Knowledge Chunks, optimized for LLM consumption and user presentation.

This process, as illustrated in Fig. 3, enables robust and intelligent query understanding, particularly critical in domains such as healthcare and technical research where both language variability and terminology accuracy are crucial.

C. Response Generation: Synthesizing the Final Answer

The Response Generation module represents the final and most critical stage of the pipeline, wherein the system synthesizes a coherent, accurate, and contextually relevant response based on the previously retrieved knowledge chunks (see Fig. 4).

This stage leverages state-of-the-art Large Language Models (LLMs) deployed via the Ollama framework, with a primary focus on LLaMA2 and Mistral.

The process starts by taking the top-ranked document chunks from the retrieval phase (usually the top three) and concatenating them into a single block of text. This text forms the context that is then paired with the user’s query in a carefully engineered prompt. The prompt begins with the instruction: “You are an expert assistant. Given the following context, answer the question accurately:”. This forces the

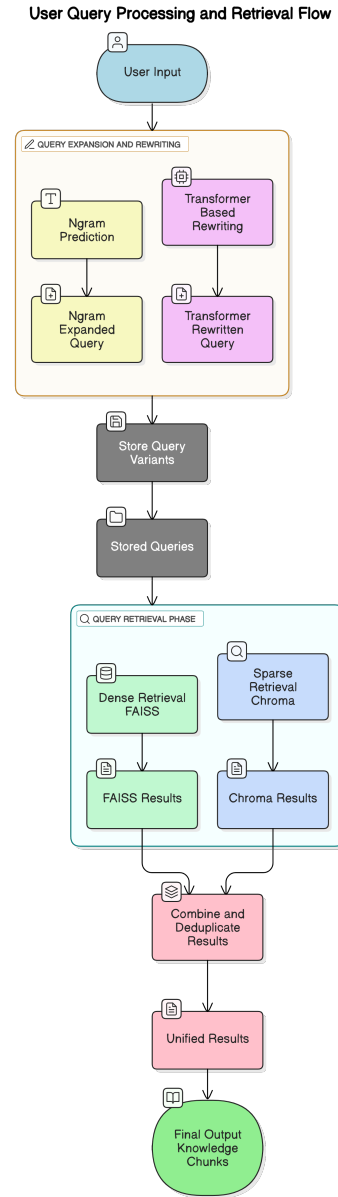


Fig. 3. query processing and answer generation

model to ground its response in the provided information, dramatically reducing errors and preventing it from generating fabricated content.

Next, the system sends this prompt to both LLaMA2 and Mistral. Each model generates its own independent answer based on the exact same context and query. This multi-model approach is a key feature of our system, as it allows for a side-by-side comparison of the outputs, which is invaluable for quality assurance. This flexibility is especially important in a sensitive domain like medicine, where subtle differences in wording can have a significant impact on the quality and safety of the answer. By comparing how different LLMs interpret the same information, we can either select the best response or present multiple perspectives to the user. As per Fig. 4

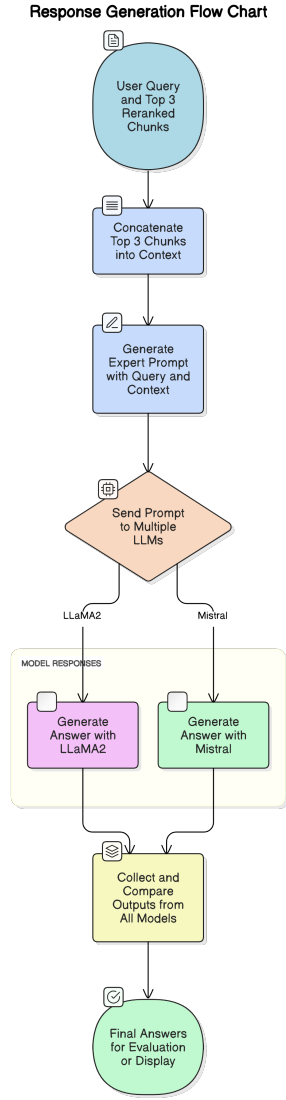


Fig. 4. Response Generation

structured process of prompting, context conditioning, and model comparison ensures that the final answer is not only contextually grounded but also medically relevant and easily understandable.

IV. RESULTS AND DISCUSSIONS

This section offers an in-depth evaluation of the PDF-based chatbot’s performance, emphasizing how well it answers questions, retrieves relevant content, handles errors, and operates under hardware limitations. The system combines document parsing, semantic search, and advanced language generation to respond to queries using content from uploaded PDFs and online web content. The chatbot leverages several key components HuggingFace embeddings, FAISS vector search, Web scraping and the LLaMA-2 language model to deliver accurate and timely answers.

A. Overview of Evaluation Strategy

Evaluating the performance of a medical chatbot involves more than verifying correctness—it requires a balanced methodology that captures lexical similarity, semantic meaning, factual accuracy, and human-like expression. To ensure robust assessment, we adopted a hybrid evaluation framework combining automated metrics, a local language model as a judge, and manual human evaluation.

Our automated evaluation employed standard NLP metrics such as BLEU, ROUGE-L, METEOR, and Cosine Similarity. These metrics allowed for quantitative comparisons between generated and reference responses. BLEU and ROUGE-L focus on n-gram overlap and structural alignment, while METEOR and Cosine Similarity capture more semantic and contextual nuance. For instance, METEOR considers word stems and synonyms, making it particularly suited for domain-specific contexts like healthcare. Cosine Similarity, using vector embeddings, effectively reflects semantic proximity even when lexical variation is high.

To complement these automated metrics, we implemented an LLM-as-a-Judge strategy using locally deployed models such as LLaMA2 or Mistral via the Ollama framework. In this setup, the local LLM is prompted with the user query, the expected reference answer, and the chatbot’s response. The model then assigns a score from 1 to 10 based on relevance, factual alignment, and completeness. This subjective assessment provides insights into aspects like nuance and coherence that are harder to quantify automatically. The LLM was integrated into the evaluation pipeline using subprocess calls to ensure streamlined execution.

The final layer of evaluation involved human judgment. Real users assessed each response across four dimensions: accuracy (especially in domain-specific queries), quality (fluency, coherence, and completeness), response time (suitability for real-time interactions), and error rate (frequency of hallucinations or factual errors). This human-in-the-loop feedback helped ensure the chatbot’s outputs were practically viable and aligned with expert expectations.

B. Experimental Evaluation Results

This section presents a detailed evaluation of the proposed PDF-based chatbot, highlighting its accuracy, response quality, retrieval performance, and robustness under standard hardware limitations. The system employs a combination of semantic search, intelligent document parsing, and generative AI to deliver meaningful responses to user queries using both static PDF content and real-time web-sourced information. By integrating HuggingFace embeddings, FAISS vector indexing, web scraping mechanisms, and the LLaMA-2 language model, the chatbot offers reliable, relevant, and context-aware answers.

The chatbot’s performance was benchmarked across a diverse set of healthcare-related questions, manually reviewed for accuracy and contextual relevance. A summary of the key evaluation metrics is presented in Table IV, including accuracy (80%), average response quality (4.2/5), average response time

Metric	Value	Description
Accuracy	80%	Percentage of correctly answered queries based on PDF content
Response Quality	4.2/5	Average user satisfaction rating for response relevance and completeness
Response Time	3.5 seconds	Average time from query submission to response generation
Error Rate	15%	Percentage of queries resulting in system errors or inadequate responses
Hardware Configuration	i5 8th Gen, 8GB RAM	Testing environment specification for performance validation

TABLE IV
SUMMARY OF KEY PERFORMANCE METRICS

(3.5 seconds), and error rate (15%). These results indicate that the majority of responses were grounded in the correct PDF-based source material and were contextually relevant. This strong performance stems from precise document chunking, semantic tokenization, and embedding techniques powered by HuggingFace. The FAISS-based retrieval layer ensures that only the top-ranked document segments are supplied to the language model, enhancing the relevance of generated answers.

C. Retrieval and Response Generation

For each query, the system retrieves the top-k relevant document chunks from the FAISS vector store, with k set to 2. This narrowed retrieval scope enhances contextual precision and boosts the effectiveness of the language model during generation. The LLaMA-2 model, interfaced through CTransformers, is used to produce answers from the retrieved data. A temperature parameter of 0.5 is used to balance deterministic accuracy with some linguistic variability, and answers are capped at 512 tokens to maintain clarity. Manual evaluations confirmed that this configuration led to consistent, coherent, and context-rich responses.

Model	Cosine Similarity	BLEU	ROUGE-L	METEOR	LLM-as-Judge
LLaMA2	0.91	0.10	0.34	0.37	8
Mistral	0.87	0.13	0.48	0.40	9

TABLE V
EVALUATION RESULTS FOR LLaMA2 AND MISTRAL ACROSS MULTIPLE METRICS

We conducted a comprehensive evaluation of the LLaMA-2 and Mistral models using five metrics: Cosine Similarity, BLEU, ROUGE-L, METEOR, and LLM-as-a-Judge. As shown in Table V and Figure V, LLaMA-2 achieved the highest Cosine Similarity score (0.91), suggesting superior semantic alignment with reference answers. However, Mistral outperformed LLaMA-2 in BLEU (0.13), ROUGE-L (0.48), METEOR (0.40), and LLM-as-a-Judge (9), indicating better surface-level and structural alignment with ground-truth responses. These results suggest that while LLaMA-2 excels in

preserving meaning, Mistral produces responses more similar in phrasing and style, which may be more favorable from a human evaluation perspective.

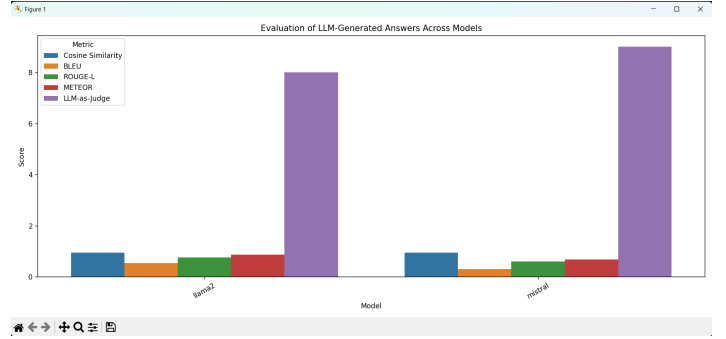


Fig. 5. Response Generation

V. CONCLUSION AND FUTURE DIRECTIONS

In this research, we developed a smart and efficient chatbot designed to answer questions using both uploaded PDF documents and live web content. The system uses a Retrieval-Augmented Generation (RAG) approach that combines strong document search techniques (dense and sparse retrieval) with advanced language models to generate high-quality answers. To improve the quality of responses, we added features like query rewriting using FLAN-T5 and n-gram-based rephrasing.

A unique feature of our system is the multi-model answer generation setup. It compares answers from models like LLaMA2 and Mistral and uses various evaluation methods such as BLEU, ROUGE-L, METEOR, and an LLM-as-a-Judge scoring system. Results show that LLaMA2 performs better in terms of meaning and context, while Mistral gives more accurate wording. The chatbot runs smoothly on a Chainlit interface, where users can upload documents, ask questions, and instantly see responses with performance scores.

This paper shows that building a powerful and reliable medical chatbot is possible even without high-end cloud resources. In the future, we plan to make the chatbot even better by training the models on specialized medical datasets, connecting it with medical knowledge bases like UMLS, and using user feedback to improve answers over time. We also aim to add features like hallucination detection, better reranking of search results (e.g., monoT5), and graph-based search to make the system more accurate and trustworthy for real-world medical use.

REFERENCES

- [1] D. Lin, "Revolutionizing retrieval-augmented generation with enhanced pdf structure recognition," *arXiv preprint arXiv:2401.12599*, 2024.
- [2] A. A. Khan, M. T. Hasan, K. K. Kemell, J. Rasku, and P. Abrahamsson, "Developing retrieval augmented generation (rag) based llm systems from pdfs: an experience report," *arXiv preprint arXiv:2410.15944*, 2024.
- [3] P. Mandikal and R. Mooney, "Sparse meets dense: A hybrid approach to enhance scientific document retrieval," *arXiv preprint arXiv:2401.04055*, 2024.

- [4] H. Zhang, J. Liu, Z. Zhu, S. Zeng, M. Sheng, T. Yang, G. Dai, and Y. Wang, "Efficient and effective retrieval of dense-sparse hybrid vectors using graph-based approximate nearest neighbor search," *arXiv preprint arXiv:2410.20381*, 2024.
- [5] S.-C. Lin and J. Lin, "A dense representation framework for lexical and semantic matching," *ACM Transactions on Information Systems*, vol. 41, no. 4, pp. 1–29, 2023.
- [6] X. Ma, Y. Gong, P. He, H. Zhao, and N. Duan, "Query rewriting in retrieval-augmented large language models," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5303–5315, 2023.
- [7] Ö. Hakdağlı, "Hybrid question-answering system: A faiss and bm25 approach for extracting information from technical document," *Orclever Proceedings of Research and Development*, vol. 5, no. 1, pp. 226–237, 2024.
- [8] J. Jin, X. Li, G. Dong, Y. Zhang, Y. Zhu, Y. Wu, Z. Li, Q. Ye, and Z. Dou, "Hierarchical document refinement for long-context retrieval-augmented generation," *arXiv preprint arXiv:2505.10413*, 2025.
- [9] J. Zhou *et al.*, "Machine learning for multi-modal pdf document interaction," *Journal of Applied AI*, 2021.
- [10] X. Wang and Y. Zhao, "Vector-based retrieval for pdf chatbot systems," *Journal of Information Retrieval Technology*, 2023.
- [11] Q. Wang *et al.*, "Enhancing pdf-based chatbot interactions with llama2," in *Proceedings of the NLP and AI Conference*, 2023.
- [12] M. R. Kabir, R. M. Sultan, F. Rahman, M. R. Amin, S. Momen, N. Mohammed, and S. Rahman, "Legalrag: A hybrid rag system for multilingual legal information retrieval," *arXiv preprint arXiv:2504.16121*, 2025.
- [13] R. Kalra, Z. Wu, A. Gulley, A. Hilliard, X. Guan, A. Koshiyama, and P. Treleven, "Hypa-rag: A hybrid parameter adaptive retrieval-augmented generation system for ai legal and policy applications," *arXiv preprint arXiv:2409.09046*, 2024.
- [14] Y. Wang, R. Ren, J. Li, W. X. Zhao, J. Liu, and J.-R. Wen, "Rear: A relevance-aware retrieval-augmented framework for open-domain question answering," *arXiv preprint arXiv:2402.17497*, 2024.
- [15] R. Baeza-Yates, B. Ribeiro-Neto, *et al.*, *Modern information retrieval*, vol. 463. ACM press New York, 1999.
- [16] J. Devlin, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
- [17] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*, pp. 8748–8763, PMLR, 2021.
- [18] M. M. L. Godinho, *MGI*. PhD thesis, Universidade Nova de Lisboa.
- [19] X. Xie, Z. Li, and T. Wu, "Ocr-enhanced chatbots for interacting with scanned pdf documents," *Journal of Document Engineering*, 2020.
- [20] J. Yang *et al.*, "Langchain for multi-source document-based chatbots," *AI and Machine Learning Review*, 2024.
- [21] R. Johnson *et al.*, "Faiss: A library for efficient similarity search of high-dimensional vectors," *Facebook AI Research*, 2020.
- [22] A. Sar, T. Choudhury, S. Sati, P. Joshi, S. Aich, B. Pant, and B. K. Dewangan, "Based chatbot development using llama2 and langchain: Training and deployment for document interaction," in *2024 OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 4.0*, pp. 1–6, IEEE, 2024.